

AMENDMENTS TO THE CLAIMS

1 1. (Currently Amended) A system for managing data transactions between a
2 first bus and a second bus, comprising:
3 a first transaction conversion module operably ~~connected~~ coupled to said first bus,
4 said first transaction conversion module being operable to receive
5 transactions from said first bus in a first format and to convert said
6 transactions into an internal format;
7 a ~~fully~~ programmable ordering rules logic module operably ~~connected~~ coupled to
8 said first transaction module to receive said converted transactions in said
9 internal format and to control issuing of said transactions in accordance
10 with a dependency relationship between individual converted transactions
11 and further operable to issue validated transactions through a plurality of
12 virtual channels using an inter-virtual channel arbiter and a plurality of
13 intra-virtual channel arbiters; and
14 a second transaction conversion module operably ~~connected~~ coupled to said
15 ~~transaction~~ ordering logic module and to said second bus, said second
16 transaction conversion module being operable to convert said validated
17 transactions into a second format for said second bus.

1 2. (Original) The system of claim 1, wherein transactions on said first
2 bus are managed using a first set of ordering rules and transactions on said second bus are
3 managed using a second set of ordering rules.

1 3. (Previously Presented) The system of claim 1, wherein said
2 transactions comprise a time stamp and wherein said ordering rules logic module is
3 operable to use said time stamp to issue said validated transactions.

1 4. (Original) The system of claim 3, wherein said rules logic module is
2 operable to validate said transactions using a protocol based on an efficiency algorithm
3 optimizing the availability of said second bus to accept a validated transaction.

1 5. (Original) The system of claim 4, wherein said ordering rules logic
2 module is programmed by a configuration status register.

1 6. Canceled

1 7. Canceled

1 8. Canceled.

1 9. Canceled.

1 10. Canceled.

1 11 Canceled.

1 12. Canceled

1 13. Canceled.

1 14. (Currently Amended) A method for managing data transactions between a
2 first bus and a second bus, comprising:

3 receiving a first transaction in a conversion module operably ~~connected~~ coupled
4 to said first bus, said first transaction conversion module being operable to
5 receive transactions from said first bus in a first format and to convert said
6 transactions into an internal format;

7 receiving said converted transaction in a fully programmable ordering rules logic
8 module operably ~~connected~~ coupled to said first transaction module;

9 using said ordering rules logic module to validate said converted transactions and
10 to control issuing of validated transactions to a second transaction module
11 in accordance with a dependency relationship between a plurality of
12 transactions, wherein said validated transactions are issued through a
13 plurality of virtual channels using an inter-virtual channel arbiter and a
14 plurality of intra-virtual channel arbiters; and

15 using a second transaction conversion module to convert said validated
16 transactions into a second format for said second bus.

1 15. (Original) The method of claim 14, wherein transactions on said first
2 bus are managed using a first set of ordering rules and transactions on said second bus are
3 managed using a second set of ordering rules.

1 16 (Previously Presented) The method of claim 15, wherein said
2 transactions comprise a time stamp and wherein said ordering rules logic module is
3 operable to use said time stamp to issue said validated transactions.

1 17. (Original) The method of claim 16, wherein said rules logic module is
2 operable to validate said transactions using a protocol based on an efficiency algorithm
3 optimizing the availability of said second bus to accept a validated transaction.

1 18. (Original) The method of claim 17, wherein said ordering rules logic
2 module is programmed by a configuration status register.

1 19. Canceled.

1 20. Canceled.

1 21. Canceled.

1 22 Canceled.

1 23. Canceled.

1 24. Canceled.

1 25. Canceled.

1 26. Canceled.